

CLAIMS

We claim:

1. A method comprising:

generating an offset map to indicate the exact location of each file in a backup image, before any file data has been written to the backup image; and

writing the offset map to the backup image before writing the file data to the backup image.

2. A method as recited in claim 1, further comprising:

subsequent to writing the offset map to the backup image, writing the files to the backup image.

3. A method as recited in claim 2, further comprising:

prior to generating the offset map, receiving a request to generate the backup image, the request indicating a backup path to be included in the backup image;

generating a list of files to be included in the backup image based on the backup path indicated in the request; and

writing the list of files to be included in the backup image to the backup image.

4. A method as recited in claim 3, wherein said generating an offset map to indicate the exact location of each file in a backup image includes calculating the

space required in the backup image for each file included in the list of files to be included in the backup image.

5. A method as recited in claim 4, wherein said calculating the space required in the backup image for each file included in the list of files to be included in the backup image includes compensating for any holes that may exist in the file, as the file is stored on a file system.

6. A method as recited in claim 4, wherein said calculating the space required in the backup image for each file included in the list of files to be included in the backup image includes compensating for any headers to be written as part of the backup image for the file.

7. A method as recited in claim 3, further comprising:

for each file in the list of files to be included in the backup image, determining whether the file has any associated streams and, if so, adding the associated streams to the list of files to be included in the backup image.

8. A method as recited in claim 3, wherein said generating a list of files to be included in the backup image based on the backup path indicated in the request includes, determining whether the backup path indicates a directory and, if so, adding any files within the directory to the list of files to be included in the backup image.

9. A method as recited in claim 3, wherein said generating a list of files to be included in the backup image based on the backup path indicated in the request includes, generating a bitmap with a number of bits equal to the total number of inodes available on the file system that hosts the backup path, each bit representing an inode associated with a file on the file system and indicating whether the associated file is included in the backup image.

10. A method as recited in claim 3, further comprising:

generating file attribute information for each file included in the list of files to be included in the backup image; and

writing the file attribute information to the beginning of the backup image on the secondary storage device.

11. A method as recited in claim 10, wherein the file attribute information includes Access Control List information.

12. A method as recited in claim 3, further comprising:

for each file in the list of files to be included in the backup image, communicating file history information to a data management application.

13. A method as recited in claim 1, wherein the offset map comprises an array having a number of elements equal to the number of inodes on the file system that hosts the backup path, each element of the array representing an inode on

the file system and indicating the exact location in the backup image of a file associated with the number of the inode.

14. A method as recited in claim 1, further comprising:

generating directory structure information based on the backup path indicated in the request; and

writing the directory structure information to the backup image;

15. A method as recited in claim 1, further comprising:

for each file in the list of files to be included in the backup image, verifying that the location in the backup image to which the file is actually written is the same location as the location of the file indicated in the offset map.

16. A method as recited in claim 1, further comprising:

communicating to a data management application whether or not the location in the backup image to which each file is actually written is the same location as the location of each file indicated in the offset map.

17. A processing system to generate a backup image, the processing system comprising:

a processor;

a network communication interface to provide the processing system with data communication with a plurality of clients, including a data management application, over a network;

a storage interface to provide the processing system with data communication with a set of mass storage devices; and

a memory containing code which, when executed by the processor, causes the processing system to execute a process of generating a backup image on behalf of the data management application, the process comprising:

generating an offset map to indicate the exact location of each file in a backup image, before any file data has been written to the backup image; and

writing the offset map to the backup image before writing the file data to the backup image.

18. A processing system as recited in claim 17, wherein the process further comprises:

subsequent to writing the offset map to the backup image, writing the files to the backup image.

19. A processing system as recited in claim 18, wherein the process further comprises:

prior to generating the offset map, receiving a request to generate the backup image, the request indicating a backup path to be included in the backup image;

generating a list of files to be included in the backup image based on the backup path indicated in the request; and

writing the list of files to be included in the backup image to the backup image.

20. A processing system as recited in claim 19, wherein said generating an offset map to indicate the exact location of each file in a backup image includes calculating the space required in the backup image for each file included in the list of files to be included in the backup image.

21. A processing system as recited in claim 20, wherein said calculating the space required in the backup image for each file included in the list of files to be included in the backup image includes compensating for any holes that may exist in the file, as the file is stored on a file system.

22. A processing system as recited in claim 20, wherein said calculating the space required in the backup image for each file included in the list of files to be included in the backup image includes compensating for any headers to be written to the backup image for the file.

23. A processing system as recited in claim 19, wherein the process further comprises:

for each file in the list of files to be included in the backup image, determining whether the file has any associated streams and, if so, adding the associated streams to the list of files to be included in the backup image.

24. A processing system as recited in claim 19, wherein said generating a list of files to be included in the backup image based on the backup path indicated in the request includes, determining whether the backup path indicates a directory and, if so, adding any files within the directory to the list of files to be included in the backup image.

25. A processing system as recited in claim 19, wherein said generating a list of files to be included in the backup image based on the backup path indicated in the request includes, generating a bitmap with a number of bits equal to the total number of inodes available on the file system that hosts the backup path, each bit representing an inode associated with a file on the file system and indicating whether the associated file is included in the backup image.

26. A processing system as recited in claim 19, wherein the process further comprises:

generating file attribute information for each file included in the list of files to be included in the backup image; and

writing the file attribute information to the beginning of the backup image on the secondary storage device.

27. A processing system as recited in claim 26, wherein the file attribute information includes Access Control List information.
28. A processing system as recited in claim 19, wherein the process further comprises:
- for each file in the list of files to be included in the backup image, communicating file history information to a data management application.
29. A processing system as recited in claim 17, wherein the offset map comprises an array having a number of elements equal to the number of inodes on the file system that hosts the backup path, each element of the array representing an inode on the file system and indicating the exact location in the backup image of a file associated with the number of the inode.
30. A processing system as recited in claim 17, wherein the process further comprises:
- generating directory structure information based on the backup path indicated in the request; and
 - writing the directory structure information to the backup image;
31. A processing system as recited in claim 17, further comprising:
- for each file in the list of files to be included in the backup image, verifying that the location in the backup image to which the file is actually written is the same location as the location of the file indicated in the offset map.

32. A processing system as recited in claim 17, wherein the process further comprises:

communicating to a data management application whether or not the location in the backup image to which each file is actually written is the same location as the location of each file indicated in the offset map.

33. A method comprising:

for each file in a list of files to be restored, reading an offset value from an offset map in a backup image, the offset value indicating the location of a file in the backup image; and

seeking to the location in the backup image where the file is located, reading the file, and writing the file to a file system.

34. A method as recited in claim 33, further comprising:

prior to reading the offset values from the offset map, receiving a request to restore files and/or directories from a backup image to a file system, the request indicating a selection of files and/or directories to restore; and

generating the list of files to be restored based on the request to restore files and/or directories.

35. A method as recited in claim 34, wherein the request includes an nlist, the nlist comprising a list of inodes corresponding to files and/or directories in a backup image.

36. A method as recited in claim 35, wherein said generating the list of files to be restored based on the request to restore files and/or directories includes, for every inode in the nlist that corresponds with a file, determining whether the corresponding file has any associated streams and, if so, adding the streams to the list of files to be restored.

37. A method as recited in claim 35, wherein said generating the list of files to be restored based on the request to restore files and/or directories includes, for every inode in the nlist, determining whether the inode corresponds with a directory and, if so, adding all files located within the directory to the list of files to be restored.

38. A method as recited in claim 34, further comprising:

for each file in the list of files to be restored, reading file attribute information associated with the file from the backup image; and

writing the file attribute information to the file system;

39. A method as recited in claim 38, wherein the file attribute information includes Access Control List information.

40. A method as recited in claim 33, further comprising:

determining, based on the offset values, whether multiple files in the list of files to be restored should be read from the backup image and written to the file system without an intervening seek operation.

41. A method as recited in claim 40, wherein determining, based on the offset values, whether multiple files in the list of files to be restored should be read from the backup image and written to the file system without an intervening seek operation includes comparing the predicted time required to seek to the location of a file with the predicted time required to read and discard one or more files in the backup image not included in the list of files to be restored.

42. A method as recited in claim 33, wherein multiple files are read from the backup image and written to the file system without an intervening seek operation only if the predicted time required to seek to a file is greater than the predicted time required to read and discard a file in the backup image that is not included in the list of files to be restored.

43. A server comprising:

a processor;

a network communication interface to provide the server with data communication with a plurality of clients, including a data management application, over a network;

a storage interface to provide the server with data communication with a set of mass storage devices; and

a memory containing code which, when executed by the processor, causes the server to execute a process of generating a backup image on behalf of the data management application, the process comprising:

for each file in a list of files to be restored, reading an offset value from an offset map in a backup image, the offset value indicating the location of a file in the backup image; and

seeking to the location in the backup image where the file is located, reading the file, and writing the file to a file system.

44. A server as recited in claim 43, wherein the process further comprises:

prior to reading the offset values from the offset map, receiving a request to restore files and/or directories from a backup image to a file system, the request indicating a selection of files and/or directories to restore; and

generating the list of files to be restored based on the request to restore files and/or directories.

45. A server as recited in claim 44, wherein the request includes an nlist, the nlist comprising a list of inodes corresponding to files and/or directories in a backup image.

46. A server as recited in claim 45, wherein said generating the list of files to be restored based on the request to restore files and/or directories includes, for every inode in the nlist that corresponds with a file, determining whether the

corresponding file has any associated streams and, if so, adding the streams to the list of files to be restored.

47. A server as recited in claim 45, wherein said generating the list of files to be restored based on the request to restore files and/or directories includes, for every inode in the nlist, determining whether the inode corresponds with a directory and, if so, adding all files located within the directory to the list of files to be restored.

48. A server as recited in claim 44, wherein the process further comprises:

for each file in the list of files to be restored, reading file attribute information associated with the file from the backup image; and

writing the file attribute information to the file system;

49. A server as recited in claim 48, wherein the file attribute information includes Access Control List information.

50. A server as recited in claim 43, wherein the process further comprises:

determining, based on the offset values, whether multiple files in the list of files to be restored should be read from the backup image and written to the file system without an intervening seek operation.

51. A server as recited in claim 50, wherein determining, based on the offset values, whether multiple files in the list of files to be restored should be read from the backup image and written to the file system without an intervening seek operation includes comparing the predicted time required to seek to the location of a file with the predicted time required to read and discard one or more files in the backup image not included in the list of files to be restored.

52. A server as recited in claim 43, wherein multiple files are read from the backup image and written to the file system without an intervening seek operation only if the predicted time required to seek to a file is greater than the predicted time required to read and discard a file in the backup image that is not included in the list of files to be restored.